Asociación Argentina
de Mecánica Computacional

Mecánica Computacional Vol XXVII, págs. 471-488 (artículo completo)
Alberto Cardona, Mario Storti, Carlos Zuppa. (Eds.)
San Luis, Argentina, 10-13 Noviembre 2008

AN ENRICHED PRESSURE INTERPOLATION FINITE ELEMENT MODEL FOR MOULD FILLING PROBLEMS

Herbert Coppola-Owen and Ramon Codina

International Center for Numerical Methods in Engineering (CIMNE), Universitat Politècnica de Catalunya, Campus Nord UPC, 08034 Barcelona, Spain, howen@cimne.upc.es, ramon.codina@upc.edu, http://www.cimne.upc.es

Keywords: Incompressible two–phase flows; stabilized finite element methods; level set; enriched shape functions; mould filling.

Abstract. Mould filling problems with slow inlet velocities lead to complex two phase flow simulations. As the Froude number decreases the coupling between the position of the inteface and the resulting flow increases. For such flows we have developed a model that enriches the finite element pressure shape functions in elements cut by the interface so that a discontinuous pressure gradient can be represented. In this work we show that the model can be applied to complex mould filling problems obtained directly from the foundry.

1 INTRODUCTION

Flows with moving interfaces (free surface and two–fluid interface problems) appear in numerous engineering applications. In this work we shall concentrate on mould filling problems but the methodology can also be applied to other kind of problems. CFD approaches for such problems can be categorized into two main groups: fixed mesh or interface capturing techniques and moving mesh or interface tracking techniques.

In interface capturing techniques a fixed computational domain is used and an interface function is used to capture the position of the interface. The interface is captured within the resolution of the fixed mesh and the boundary conditions at the interface are somehow approximated.

In interface tracking techniques the mesh is updated in order to track the interface. The simplest approach is to deform the mesh without changing its topology, but it is valid only for very simple flows. As the flow becomes more complex and unsteady remeshing and consequently the projection of the results from the old to the new mesh are needed. In 3D calculations, these operations can introduce costs that can render moving mesh techniques unfeasible. We will therefore use interface capturing techniques.

Contrary to what one might intuitively think, we have observed that in mould filling problems, lower filling velocities typically lead to more complex simulations. That is to say, low Froude number flows pose special difficulties for two phase flows. The lower the Froude number, the higher the importance of the gravitational forces. Since the spatial distribution of the gravitational forces is determined by the position of the interface, the coupling between the position of the interface and the resulting flow increases as the Froude number decreases. An accurate representation of the pressure in the elements cut by the interface is needed for such flows. By enriching the pressure finite element shape functions we have obtained important improvements in simple examples. In this work we extend the application of the model to real mould filling problems.

The idea of enriching the representation of an unknown at a material discontinuity is not new and several approaches can be found in the literature (see Chessa and Belytschko (2003); Minev et al. (2003)). In Coppola-Owen and Codina (2005) we introduced the model we will use in this paper and compared it against the previously cited methods.

Fixed mesh methods generally share two basic steps, one where the motion in both phases is found as the solution of the Navier–Stokes equations for a one phase flow with variable properties and the other one, where an equation for an interface function that allows to determine the position of the interface, and thus the properties to be assigned in the previous step, is solved. The different methods differ mainly in the method used to determine the position of the interface but also differences can be found in the way to approximate the properties to be used close to the interface.

Referring to the evolution of the fluid interface, we update it using the so called *Level Set* method (see Chang et al. (1996); Sussman et al. (1999) and Osher and Fedkiw (2001) for an overview), also called *pseudo–concentration technique* (Thompson (1986)) and very similar to the volume of fluid (VOF) technique (Hirt and Nichols (1981)). This formulation has been widely used to track free surfaces in mould filling (see for example Codina et al. (1994); Lewis et al. (1995), among other references) and other metal forming processes.

The Level Set method leads to a transport partial differential equation whose solution determines the position of the free surface as an isovalue of the unknown of this equation, which we will call ψ. This equation is hyperbolic and therefore it is also necessary to use a stabilized finite element method to solve it. The enrichment we introduce in this paper does not depend

on the approach used to capture the interface.

A stabilized finite element method with standard trapezoidal rule time discretization is used to solve both the Level Set equation and the two fluid Navier Stokes equations. SUPG stabilization is applied to the Level Set equation. The Navier Stokes equations are stabilized using the Algebraic Sub-Grid Scales (ASGS) (Codina (2001)) method that deals with convection-dominated flows and allows equal velocity-pressure interpolations (thus avoiding the need to satisfy the classical inf-sup condition). Equal order linear tetrahedral finite elements are used to discretize the complex tridimensional geometries found in mould filling problems. In the elements cut by the interface the P1 (linear simplicial elements) pressure shape functions are supplemented with an additional shape function that is zero at all the element nodes, continuous within the element and has a constant gradient on each side of the interface. This shape function is local to each element and the corresponding degree of freedom can therefore be condensed prior to assembly, making the implementation quite simple on any existing finite element code. The details will be discussed in Section 4.

The remainder of the paper is organized as follows. In Section 2 we describe the mathematical model used to solve Navier–Stokes equations when no enrichment functions are used and in Section 3 we briefly describe the Level Set Method used. In Section 4 we present the enrichment functions used and some implementation details. Finally in Section 5 we present straight out of the foundry examples. The results can be used to improve the casting process. Regions with high velocities that can lead to premature wear of the mould can be predicted. The quality of the resulting piece can also be improved, for example, by determining regions of possible air entrapment.

2 FINITE ELEMENT APPROXIMATION OF THE TWO–FLUID NAVIER STOKES EQUATIONS

The velocity and pressure fields of two incompressible fluids moving in the domain $\Omega = \Omega_1 \cup \Omega_2$ during the time interval (t_0, t_f) can be described by the incompressible two–fluid Navier–Stokes equations:

$$\rho\Big[\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u}\Big] - \nabla\cdot[2\mu\boldsymbol{\varepsilon}(\boldsymbol{u})] + \nabla p = \boldsymbol{f}, \tag{1}$$

$$\nabla\cdot\boldsymbol{u} = 0, \tag{2}$$

where ρ is the density, $\boldsymbol{u}$ the velocity field, μ the dynamic viscosity, p the pressure, $\boldsymbol{\varepsilon}(\cdot)$ the symmetric gradient operator and $\boldsymbol{f}$ the vector external body forces, which includes the gravity force $\rho\boldsymbol{g}$ and buoyancy forces, if required. The density, velocity, dynamic viscosity and pressure are defined as

$$\boldsymbol{u},\ p,\ \rho,\ \mu = \begin{cases} \boldsymbol{u}_1\ p_1,\ \rho_1,\ \mu_1 & \boldsymbol{x}\in\Omega_1, \\ \boldsymbol{u}_2\ p_2,\ \rho_2,\ \mu_2 & \boldsymbol{x}\in\Omega_2, \end{cases}$$

where Ω_1 indicates the part of Ω occupied by fluid number 1 and Ω_2 indicates the part of Ω occupied by fluid number 2. The extent of Ω_1 and Ω_2 is given by the Level Set function ψ.

Let $\boldsymbol{\sigma}$ be the stress tensor and $\boldsymbol{n}$ the unit outward normal to the boundary $\partial\Omega$. Denoting by an over-bar prescribed values, the boundary conditions to be considered are:

$$\boldsymbol{u} = \overline{\boldsymbol{u}} \text{ on } \Gamma_{\mathrm{du}}, \quad \boldsymbol{n}\cdot\boldsymbol{\sigma} = \boldsymbol{0} \text{ on } \Gamma_{\mathrm{nu}}, \quad \boldsymbol{u}\cdot\boldsymbol{n} = 0, \quad \boldsymbol{n}\cdot\boldsymbol{\sigma}\cdot\boldsymbol{g}_1 = \tau_{w1}, \quad \boldsymbol{n}\cdot\boldsymbol{\sigma}\cdot\boldsymbol{g}_2 = \tau_{w2} \text{ on } \Gamma_{\mathrm{mu}},$$

for $t \in (t_0, t_f)$. Vectors $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$ (for the three-dimensional case) span the space tangent to Γ_{mu}. τ_{w1} and τ_{w2} are the tractions due to the wall law discussed briefly at the end of this Section.

Observe that $\Gamma_{\rm du}$ is the part of the boundary with Dirichlet velocity conditions, $\Gamma_{\rm nu}$ the part with Neumann conditions (prescribed stress) and $\Gamma_{\rm mu}$ the part with mixed conditions. These three parts do not intersect and are a partition of the whole boundary $\partial\Omega$. Initial conditions have to be appended to the problem.

ASGS (Codina (2001)) stabilization is used to deal with convection-dominated flows and to circumvent the well known div-stability restriction for the velocity and pressure finite element spaces (Brezzi and Fortin (1991)), allowing in particular equal interpolation for both unknowns.

Let $\boldsymbol{V}_h^*$ and Q_h^* be the finite element spaces to interpolate vector and scalar functions, respectively, constructed in the usual manner and using the same interpolation from a finite element partition $\Omega = \bigcup \Omega^e$, $e = 1, ..., n_{\rm el}$, where $n_{\rm el}$ is the number of elements. From these spaces one can construct the subspaces $\boldsymbol{V}_{h,u}$ and Q_h for the velocity and the pressure, respectively. The former incorporates the Dirichlet conditions for the velocity components and the latter has one pressure fixed to zero if the normal component of the velocity is prescribed on the whole boundary. The space of velocity test functions, denoted by $\boldsymbol{V}_h$, is constructed as $\boldsymbol{V}_{h,u}$ but with functions vanishing on the Dirichlet boundary. Let also θ, with $0 < \theta \leq 1$, be the parameter of the trapezoidal rule for time discretisation and δt the time step size, for simplicity constant. The algorithmic solution to the problem will be computed at $t^n = n\delta t$, $n = 1, 2, ...$ The ASGS monolithic discrete problem associated with the Navier–Stokes equations (1)-(2), discretizing in time using the generalized trapezoidal rule, and linearizing the convective term using a Picard scheme, can be written as follows: Given a velocity $\boldsymbol{u}_h^n$ at time t^n and a guess for the unknowns at an iteration $i-1$ at time t^{n+1}, find $\boldsymbol{u}_h^{n+\theta,i} \in \boldsymbol{V}_{h,u}$ and $p_h^{n+\theta,i} \in Q_h$, by solving the discrete variational problem:

$$
\begin{aligned}
&\int_\Omega \rho \frac{\boldsymbol{u}_h^{n+\theta,i} - \boldsymbol{u}_h^n}{\theta\delta t} \cdot \boldsymbol{v}_h \,\mathrm{d}\Omega + \int_\Omega \rho(\boldsymbol{u}_h^{n+\theta,i-1} \cdot \nabla)\boldsymbol{u}_h^{n+\theta,i} \cdot \boldsymbol{v}_h \,\mathrm{d}\Omega \\
&\quad + \int_\Omega \mu\boldsymbol{\varepsilon}(\boldsymbol{u}_h^{n+\theta,i}) : \boldsymbol{\varepsilon}(\boldsymbol{v}_h) \,\mathrm{d}\Omega - \int_\Omega \nabla \cdot \boldsymbol{v}_h p_h^{n+\theta,i} \,\mathrm{d}\Omega - \int_\Omega \boldsymbol{v}_h \cdot \boldsymbol{f} \,\mathrm{d}\Omega \\
&\quad + \sum_{e=1}^{n_{\rm el}} \int_{\Omega^e} \tau_1^{n+\theta,i-1} \left[\frac{\mu}{\rho}\Delta\boldsymbol{v}_h + (\boldsymbol{u}_h^{n+\theta,i-1} \cdot \nabla)\boldsymbol{v}_h\right] \cdot \left[\frac{1}{\theta\delta t}(\boldsymbol{u}_h^{n+\theta,i} - \boldsymbol{u}_h^n)\right. \\
&\qquad \left. - \mu\Delta\boldsymbol{u}_h^{n+\theta,i} + \rho(\boldsymbol{u}_h^{n+\theta,i-1} \cdot \nabla)\boldsymbol{u}_h^{n+\theta,i} + \nabla p_h^{n+\theta,i} - \boldsymbol{f}\right] \mathrm{d}\Omega \\
&\quad + \sum_{e=1}^{n_{\rm el}} \int_{\Omega^e} \tau_2^{n+\theta,i-1} (\nabla \cdot \boldsymbol{v}_h)(\nabla \cdot \boldsymbol{u}_h^{n+\theta,i}) \,\mathrm{d}\Omega = 0, \qquad \forall\, \boldsymbol{v}_h \in \boldsymbol{V}_h, \\
&\int_\Omega \rho q_h \nabla \cdot \boldsymbol{u}_h^{n+\theta,i} + \sum_{e=1}^{n_{\rm el}} \int_{\Omega^e} \tau_1^{n+\theta,i} \nabla q_h \cdot \left[\frac{1}{\theta\delta t}(\boldsymbol{u}_h^{n+\theta,i} - \boldsymbol{u}_h^n)\right. \\
&\qquad \left. - \mu\Delta\boldsymbol{u}_h^{n+\theta,i} + \rho(\boldsymbol{u}_h^{n+\theta,i-1} \cdot \nabla)\boldsymbol{u}_h^{n+\theta,i} + \nabla p_h^{n+\theta,i} - \boldsymbol{f}\right] \mathrm{d}\Omega = 0\,, \quad \forall\, q_h \in Q_h,
\end{aligned}
$$

for $i = 1, 2, ...$ until convergence, that is to say, until $\boldsymbol{u}_h^{n+\theta,i-1} \approx \boldsymbol{u}_h^{n+\theta,i}$ and $p_h^{n+\theta,i} \approx p_h^{n+\theta,i-1}$ in the norm defined by the user.

The parameters τ_1 and τ_2 are chosen in order to obtain a stable numerical scheme with optimal convergence rates (see Codina (2001) and references therein for details). They are computed within each element domain Ω^e. We take them as:

$$
\tau_1 = \frac{\rho(h^e)^2}{4\mu + 2\rho h^e|\boldsymbol{u}^e|} \quad \text{and} \quad \tau_2 = \mu + \frac{1}{2}\rho h^e|\boldsymbol{u}^e|,
$$

where h^e and $|\boldsymbol{u}^e|$ are a typical length and a velocity norm of element e, respectively.

Once the algorithm has produced a converged solution, the velocity field at t^{n+1} can be updated from the velocity at $t^{n+\theta}$ by using the relation $\boldsymbol{u}^{n+1} = [\boldsymbol{u}^{n+\theta} - (1-\theta)\boldsymbol{u}^n]/\theta$.

The material properties (μ, ρ) in the elements cut by the interface are taken at each integration point (k) as (μ_1, ρ_1) or (μ_2, ρ_2) depending on the values of ψ_k.

The enrichment technique presented in Section 4 can be understood as a modification of the pressure space Q_h to $\hat{Q}_h$, with $Q_h \subset \hat{Q}_h$. Apart from this, the resulting formulation follows exactly the previous setting.

Since the problems we have solved are highly turbulent the viscosity in the previous equations has been calculated as $\mu = \mu_L + \mu_T$, where μ_L is the molecular, constant, viscosity and $\mu_T = \mu_T(\boldsymbol{u})$ is the additional turbulent viscosity defined by

$$\mu_T = \rho\, C\, h^2 \sqrt{2\, \varepsilon\,(u) : \varepsilon\,(u)},$$

where h is the size of the element where it is computed and C is a constant we have taken as $C = 0.4$.

Due to the high Reynolds number of the problems we are dealing with, no slip boundary conditions would require extremely fine meshes along the boundary that would make them computationally unfeasible. The solution we have adopted is to use wall functions (Launder and Spalding (1974)) that describe the behavior of the flow near a solid wall. The normal component of the velocity is set to zero. In the tangential direction a traction that depends on the velocity at the boundary and is opposed to the direction of the flow is applied.

$$\tau_w = -\rho \frac{u_*^2}{|\boldsymbol{u}|}\, \boldsymbol{u}$$

where u_* can be determined from the following set of equations

$$u^+ = \frac{1}{\kappa} ln\left(1 + \kappa\, y^+\right) + 7.8\left[1 - e^{-y^+/11.0} - \frac{y^+}{11.0} e^{-0.33 y^+}\right]$$

$$u^+ = \frac{\rho\,|\boldsymbol{u}|\, u_*}{\tau_w},$$

$$y^+ = \frac{\rho \delta u_*}{\mu}.$$

δ is the distance betewen the computational boundary and the wall, $\kappa = 0.41$ is the Von Karman contant and y^+ and u^+ are non dimensional distances and velocities respectivelly.

3 IMPLEMENTATION OF THE LEVEL SET METHOD

The basic idea of the Level Set method is to define a smooth scalar function, say $\psi(\boldsymbol{x}, t)$, over the computational domain Ω that determines the extent of subdomains Ω_1 and Ω_2. For instance, we may assign positive values to the points belonging to Ω_1 and negative values to the points belonging to Ω_2. The position of the fluid front will be defined by the iso-value contour $\psi(\boldsymbol{x}, t) = 0$. The evolution of the front $\psi = 0$ in any control volume $V_t \subset \Omega$ which is moving with a divergence free velocity field $\boldsymbol{u}$ leads to:

$$\frac{\partial \psi}{\partial t} + (\boldsymbol{u} \cdot \nabla)\psi = 0 \qquad \text{in } \Omega \times (t_0, t_f). \tag{3}$$

This equation is hyperbolic and therefore boundary conditions for ψ have to be specified at the inflow boundary, defined as:

$$\Gamma_{\text{inf}} := \{\boldsymbol{x} \in \partial\Omega \mid \boldsymbol{u} \cdot \boldsymbol{n} < 0\}.$$

Function ψ is the solution of the hyperbolic equation (3) with the boundary conditions:

$$\begin{aligned} &\psi = \overline{\psi} && \text{on } \Gamma_{\text{inf}} \times (t_0, t_f), \\ &\psi(\boldsymbol{x}, 0) = \psi_0(\boldsymbol{x}) && \text{in } \Omega. \end{aligned}$$

The initial condition ψ_0 is chosen in order to define the initial position of the fluid front to be analyzed. The boundary condition $\overline{\psi}$ determines which fluid enters through a certain point of the inflow boundary.

Due to the pure convective type of the equation for ψ, we use the SUPG technique for the spatial discretisation. Again, the temporal evolution is treated via the standard trapezoidal rule.

For the numerical solution of the Level Set equation it is preferable to have a function without large gradients. Since the only requirement such a function must meet is $\psi = 0$ at the interface, a signed distance function ($\mid \nabla\psi \mid \; = 1$) is used. Under the evolution of the Level Set equation, ψ will not remain a signed distance function and thus needs to be reinitialized. This can be achieved by redefining ψ for each node of the finite element mesh according to the following expression:

$$\psi = \text{sgn}(\psi^0)d, \tag{4}$$

where ψ^0 stands for the calculated value of ψ, d is the distance from the node under consideration to the front, and $\text{sgn}(\cdot)$ is the signum of the value enclosed in the parenthesis.

Since the objective of this paper is to analyze the improvements that can be obtained in the solution of the Navier Stokes equations and not to optimize the solution of the Level Set equations, a very simple algorithm has been used to calculate the distance d. Using linear elements, the free surface is approximated by triangular planes p (lines in 2D). Then the perpendicular distance d_ip of each grid point i to each plane p can be computed. The minimum distance from each nodal point to the planes is the required distance between the point and the front ($d_i = \min_{\text{p}}\{\text{d}_\text{i}\text{p}\}$).

4 DISCONTINUOUS GRADIENT PRESSURE SHAPE FUNCTIONS

In fixed grid finite element methods the whole domain Ω is subdivided into elements Ω^e. Within each element the unknowns are interpolated as

$$\phi_h|_{\Omega^e} = \sum_{I=1}^{\text{NNODE}} N_e^I \Phi_e^I,$$

where NNODE is the number of element nodes.

In typical finite element methods, ∇N_e^I are continuous within each element and therefore $\nabla\phi_h|_{\Omega^e}$ is continuous. When the interface crosses an element the discontinuity in the material properties leads to discontinuities in the gradients of the unknowns that the interpolation used cannot capture. In the case of different density fluids under gravitational forces the jump in the pressure gradient at the interface cannot be represented correctly by typical finite element functions if the elements are cut by the interface. The errors in the pressure give rise to spurious

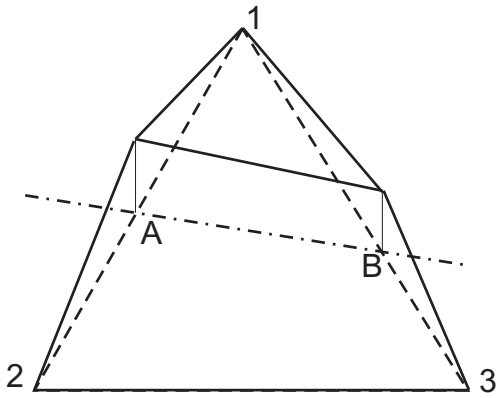

Figure 1: 2D Enrichment function for a cut element

velocities that can render the solution meaningless. Also, viscosity discontinuities can lead to discontinuous velocity gradients.

Enrichment methods add degrees of freedom at elements cut by the interface in order to reduce interpolation errors. In our particular case we add only one pressure degree of freedom per cut element. Therefore the pressure in elements cut by the interface is interpolated as

$$p_h|_{\Omega^e} = \sum_{I=1}^{\text{NNODE}} N_e^I P_e^I + N_e^{\text{ENR}} P_e^{\text{ENR}}. \tag{5}$$

The shape function N_e^{ENR} we introduce has a constant gradient on each side of the interface, its value is zero at the element nodes and is C^0 continuous in Ω^e. The added degree of freedom is local to the element and can therefore be condensed after the element matrix has been computed and before assembly. The resulting pressure finite element space is made of functions discontinuous across interelement boundaries, and thus it is a subspace of $L^2(\Omega)$, but not of $H^1(\Omega)$, as would be the case using $P1 - P1$ elements. However, our method is still conforming. If we had tried to use the previous enrichment functions for the velocity we would have obtained a non conforming method.

In Fig. 1 we show a sketch of the enrichment function we use for an element cut by the interface in the 2D case. The element has nodes named 1, 2 and 3 and the interface cuts the element edges at points A and B. A way to build such function is as follows. Suppose that node 1 belongs to Ω_1 and nodes 2 and 3 belong to Ω_2. Let $\Omega_1^e = \Omega_1 \cap \Omega^e$ and $\Omega_2^e = \Omega_2 \cap \Omega^e$. In Ω_2^e we want N^{ENR} to have constant gradient and to have a zero value at $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$. We can therefore define

$$N^{\text{ENR}}|_{\Omega_2^e} = k^1 \;\; N^1\big|_{\Omega_2^e},$$

where k^1 is a constant to be defined. By definition we want $N^{\text{ENR}}(\boldsymbol{x}_A) = 1$. As we are using linear elements to interpolate the Level Set function we have that

$$N^1(\boldsymbol{x}_A) = \frac{\Psi^2 - \psi_c}{\Psi^2 - \Psi^1},$$

where Ψ^i is the value of ψ at node i, and therefore

$$k^1 = \frac{\Psi^2 - \Psi^1}{\Psi^2 - \psi_c}.$$

Now we have k^1 we can find

$$N^{\text{ENR}}(\boldsymbol{x}_B)|_{\Omega_2^e} = k^1 \;\; N^1(\boldsymbol{x}_B)\big|_{\Omega_2^e} = k^1 \, \frac{\Psi^3 - \psi_c}{\Psi^3 - \Psi^1}.$$

We can proceed to find $N^{\text{ENR}}|_{\Omega_1^e}$. We want it to have a constant gradient in Ω_1^e and to be zero at $\boldsymbol{x}_1$. Then

$$N^{\text{ENR}}|_{\Omega_1^e} = k^2 \;\; N^2\big|_{\Omega_1^e} + k^3 \;\; N^3\big|_{\Omega_1^e}.$$

Using once more that $N^{\text{ENR}}(\boldsymbol{x}_A) = 1$ and the fact that $N^3(\boldsymbol{x}_A) = 0$ we get

$$k^2 = \frac{1}{N^2(\boldsymbol{x}_A)} = \frac{\Psi^1 - \Psi^2}{\Psi^1 - \psi_c}.$$

Since we want the enrichment function to be continuous in Ω^e we need

$$N^{\text{ENR}}(\boldsymbol{x}_B)|_{\Omega_2^e} = N^{\text{ENR}}(\boldsymbol{x}_B)|_{\Omega_1^e},$$

then, as $N^2(\boldsymbol{x}_B) = 0$,

$$k^3 = N^{\text{ENR}}(\boldsymbol{x}_B)|_{\Omega_2^e} \frac{1}{N^3(\boldsymbol{x}_B)} = k^1 \, \frac{\Psi^3 - \psi_c}{\Psi^3 - \Psi^1} \frac{\Psi^1 - \Psi^3}{\Psi^1 - \psi_c},$$

$$k^3 = -k^1 \, \frac{(\Psi^3 - \psi_c)}{(\Psi^1 - \psi_c)}.$$

We have obtained an enrichment function that is proportional to N^1 on Ω_2^e and a linear combination of N^2 and N^3 on Ω_1^e, where the values of k^1, k^2, k^3 only depend on the values of the Level Set function at the element nodes. It is very easy to obtain the enrichment function an its Cartesian derivates from the usual shape function. It seems worthwhile to remark that $N^{\text{ENR}}|_{\Omega^e}$ does not belong to the space formed by $N^1|_{\Omega^e}$, $N^2|_{\Omega^e}$, $N^3|_{\Omega^e}$. The same ideas have been used to obtain N^{ENR} for 3D elements.

In order to capture the discontinuities and take advantage of the enrichment functions used, the integration rules need to be modified in elements cut by the front. The modified integration we use consists in dividing each tetrahedral (triangular in 2D) element into up to six tetrahedral (three triangular in 2D) sub elements. For each sub element the same integration rule as for the non–cut elements is used.

When using enrichment functions for the pressure, the material properties μ, ρ are taken as μ_1, ρ_1 or μ_2, ρ_2 depending on which part of the domain (Ω_1 or Ω_2) the integration point is found.

Since the pressure space is enriched, a remark is needed concerning pressure stability. If we had used a velocity-pressure interpolation satisfying the inf-sup condition, the enrichment of the pressure could have led to an unstable velocity-pressure pair. However, we are using a stabilized finite element formulation. Even though we have no stability analysis for the enriched pressure space, we have not encountered any type of stability misbehavior.

A final remark is required concerning the extension of the proposed enrichment to higher order elements. Since the intention is to add a pressure field able to deal with discontinuous pressure gradients, but constant in each fluid phase, exactly the same methodology as described for P_1 elements can be applied to higher order elements. The construction of the enriched pressures can be based only in the linear part of the interpolation basis functions of these higher order elements. This is particularly simple when they are implemented using a hierarchical basis. The case of quadrilateral elements (or hexahedra in 3D) can be dealt with by splitting the quadrilateral intro triangles (or tetrahedra).

5 NUMERICAL EXAMPLES

In this section we present three mould filling examples borrowed directly from the foundry where the improvements obtained with the proposed formulation show up clearly. Simpler examples can be found in Coppola-Owen and Codina (2005). The results obtained with the enriched formulation are compared with those obtained with a typical finite element formulation with no enrichment. In the first and third examples the flow rate is prescribed at the inlet and in the second one the pressure (actually the stress) is prescribed. The walls of the mould are supposed impervious, that is, the normal component of the velocity is prescribed to zero. Each of the pieces has one or more air outlets where zero traction is applied.

The linear systems arising from the Navier Stokes and Level Set equations are solved with a GMRES iterative solver. The Navier Stokes solver is by far more computational demanding than the Level Set solver. It uses an ILU preconditioner with filling 10 and threshold 0.01. The Krylov dimension is 250, the maximum number of iterations is 500 and the convergence tolerance is 1.0×10^{-6} .

For the nonlinerity due the convective term, the stabilization and the turbulence model Picard iteration is used. The convergence tolerance is set to one percent variation of the solution and a maximum of seven nonlinear iterations are allowed.

The runs were performed on a PC with AMD Athlon(tm) 64 X2 Dual Core Processor 4400+ running at 2.2 GHz with 3 Gbyte of RAM using the Intel Fortran compiler under Ubuntu.

5.1 Hollow mechanical piece

The first example is a hollow mechanical piece made of steel; $\rho = 7266.0$ and $\mu = 6.7 \times 10^{-3}$ (SI units). This piece is interesting because it has relatively thin walls which make the mesh quite complex. The code is forced to obtain acceptable results with few elements in the thickness. The arrangement we simulate consists of two pieces together with the filling channel used during the actual filling process. The inlet velocity is $0.113\, m/s$ and the size of each piece is approximately $0.16 \times 0.16 \times 0.13\, m$. The whole filling process takes 16 seconds.

Two unstructured triangular meshes have been used. The coarse one has 72032 elements and 16149 nodes and the fine one has 575803 elements and 116214 nodes. They are shown in Figure 2. The time steps size used is 0.005 seconds. The Reynolds number based on the inlet velocity and the length of the filling channel is $Re = 2.45 \times 10^4$, and the Froude number is $Fr = 0.0065$.

In the Figure 3 the evolution of the interface is shown for several time steps during the filling process. In the first step the interface is still inside the filling channel. For the second one it has entered both pieces. In the third one the interface reaches the bottom of each piece. As we will comment later, this is one of the most complicated moments in the simulation. In the final figure more than half of each piece has been filled. The evolution of the front is very similar in both pieces. Despite a coarse grid has been used the evolution of the interface is captured quite satisfactorily as one can observe by comparing with the results shown for the fine mesh in Figure 4.

Knowing how the interface evolves is important during the mould design as it can be used to change the position of the inlets or alter the filling velocity to improve the quality of the resulting piece. When defects appear, having some insight on the way the flow evolves is of great help to the foundry person because it is very difficult to actually see what is happening inside the mould.

The evolution of the interface using the fine mesh in shown Figure 4. The shape of the in-

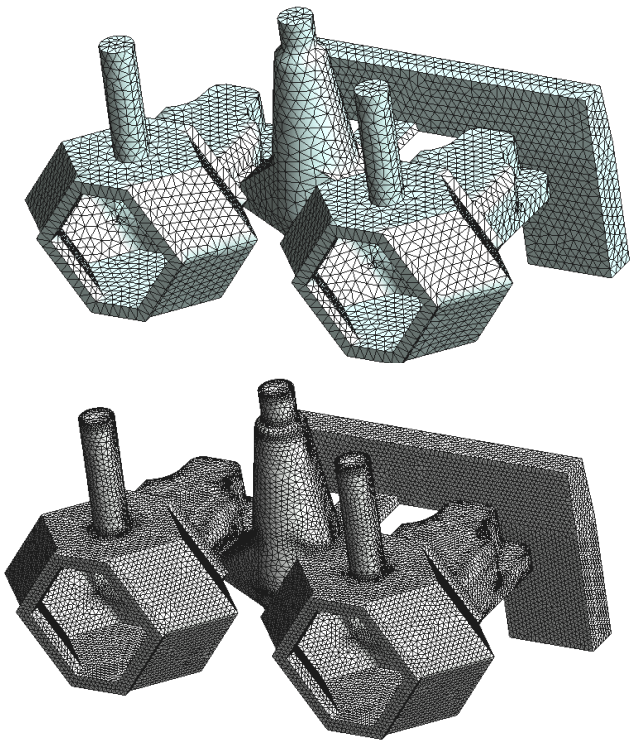

Figure 2: Coarse and fine meshes for the hollow mechanical piece

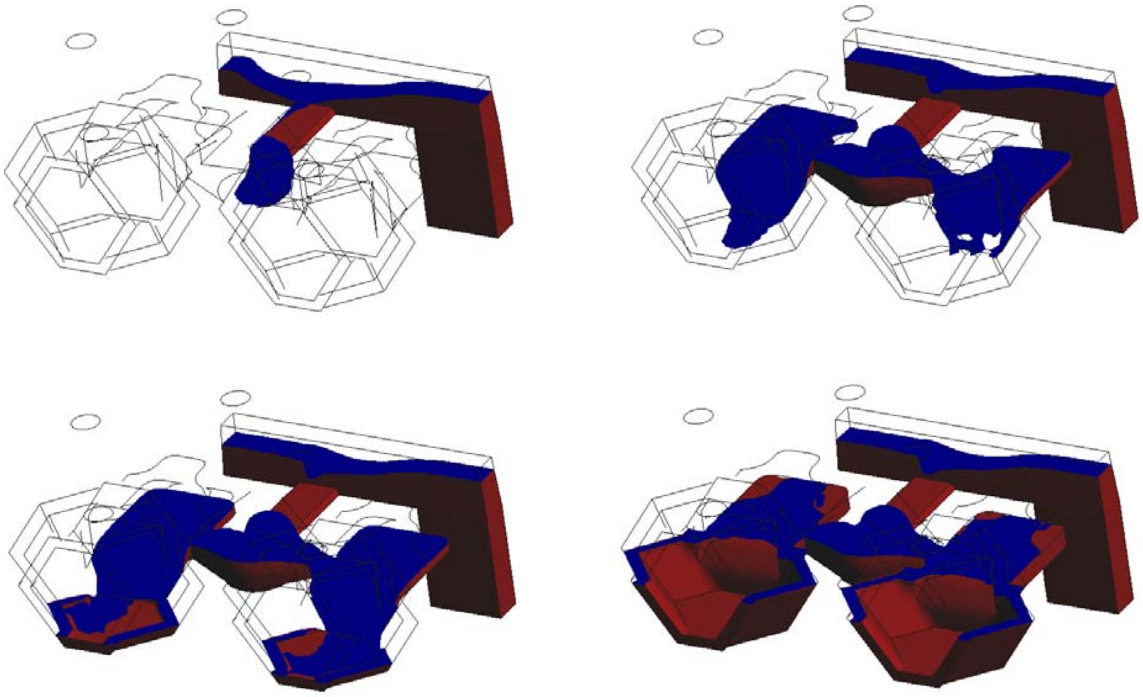

Figure 3: Interface position at $t = 1.2, 2.5, 5.0$ and 9.0 s using the coarse mesh

terface is smoother than the one obtained with the coarse mesh but there is no mayor difference in the way the flow evolves. The most noticeable change is that for each time step the results obtained with the fine mesh show a bigger percentage of filled volume. This is related to numerical mass losses and is analyzed in more detail in Figure 5. Since foundry pieces are usually complex and it is common to fill several pieces at the same time (not only two as in the example) it is important to have a code that can provide the user with acceptable results even with coarse meshes.

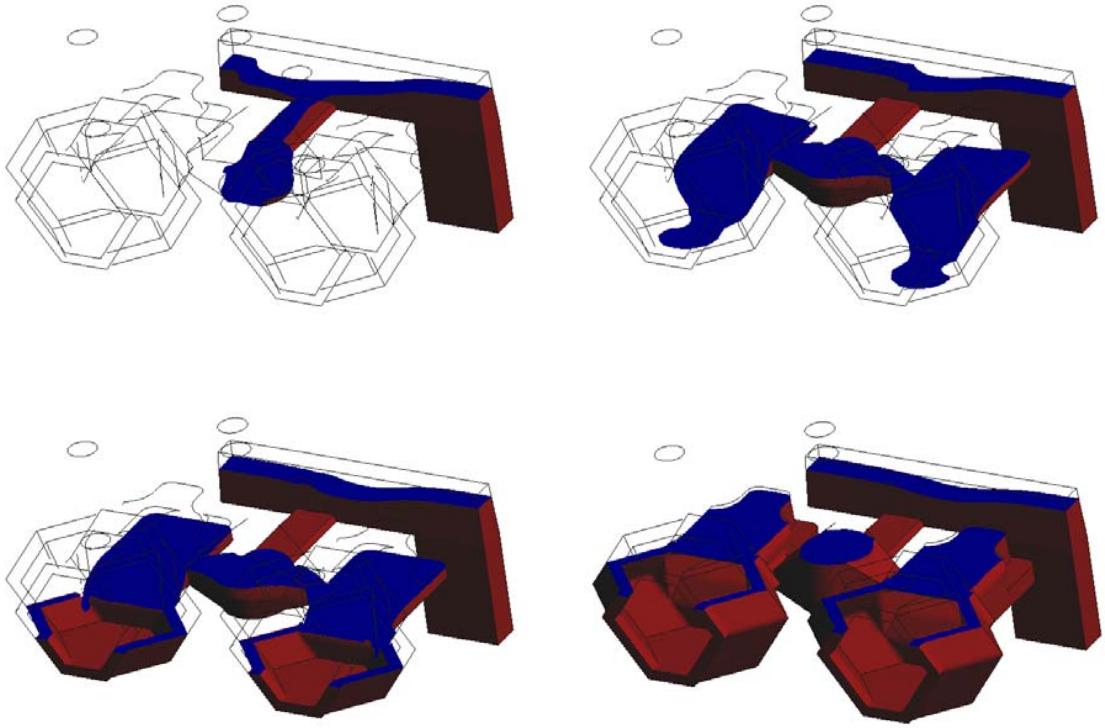

Figure 4: Interface position at $t = 1.2, 2.5, 5.0$ and $9.0\ s$ using the fine mesh

In Figure 5 we compare temporal evolution of the injected and filled volumes using both meshes. The injected volume is the same for both meshes. The difference between the filled and injected volumes is the numerical mass loss. It is reduced as the mesh is refined as one could expect. The amount of mass loss can give us some idea on the quality of our results and indicate the most complex moments during the simulation. In our example, we can see that the most important mass loss occurs when the filled volume is between $0.0004\ m^3$ and $0.0006\ m^3$. It corresponds to the moment when the bottom of each piece is being filled. This suggests that a mesh refinement close to that area might improve the solution.

The fluid mass loss comes from several sources. Despite we are solving the incompressible Navier Stokes equations the numerical results are not exactly divergence free. Since our meshes are quite coarse there will be errors in the satisfaction of both the continuity and momentum equations. Pressure stabilization also affects the satisfaction of the incompressibility condition. The errors in the transport of the Level Set function can also cause fluid loss. After solving the level set function it needs to be reinitialized; this may also introduce errors. There might also be some coupling between the previous sources of error.

The errors introduced by the reinitialization may be compensated by moving the interface

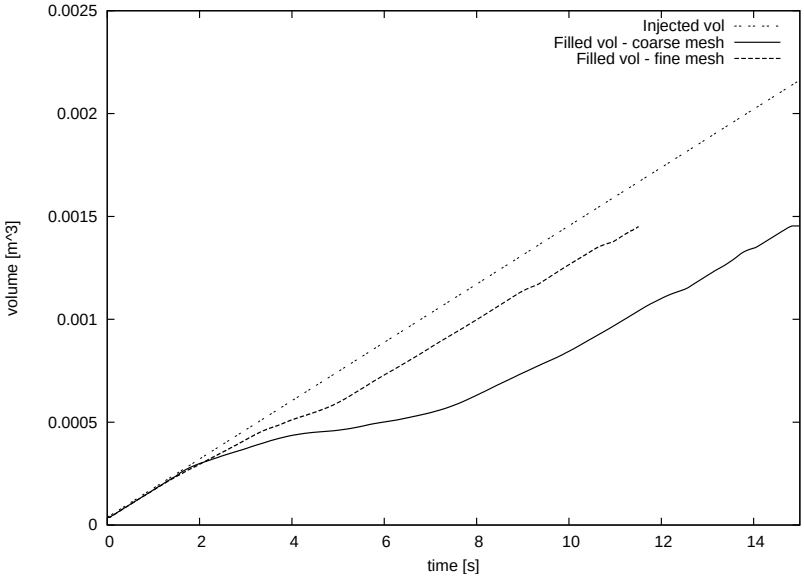

Figure 5: Filled volume vs. injected volume for both meshes

so that the volume of fluid is the same before and after the reinitialization. We have introduced this correction and we are also reinitializing the Level Set function every 4 steps in order to try to minimize mass loss.

The flow pattern during the filling process is also important to the foundry person. For example, regions of high velocities can lead to premature mould wear and should be avoided. In Figure 6 the flow field at different time steps obtained with the fine mesh is shown.

We believe that the effectiveness of the method we propose depends strongly on the pressure enrichment we introduced in Coppola-Owen and Codina (2005). In order to prove this we have run the same problem on the fine mesh without using the pressure enrichment. The results are much poorer than those shown previously. By the time the filled volume fraction reaches a 14 percent of the mould the mass loss is so important that most of the injected fluid is being lost numerically. The results lose any sense and therefore the runs was stopped. The evolution of the filled and injected volumes in shown in Figure 7.

5.2 Alloy wheel

The second example is an automotive alloy wheel. The flow is created by applying a pressure on the fluid as is done in the actual filling process for this piece. The flow rate is then determined by the resistance exerted on the fluid. We have observed the friction may be high in the vertical tube through which the molten metal is injected. Therefore, for this case, we will simulate the whole filling channel.

The pressure at the inlet varies linearly from $2.21 \times 10^4 N/m^2$ at the beginning of the simulation to $1.17 \times 10^5 N/m^2$ after 4.4 seconds. The physical properties we have used are those of aluminum, $\rho = 2700.0$ and $\mu = 1.3 \times 10^{-3}$ (SI units). The Reynolds number based on a typical velocity inside the wheel ($0.5\ m/s$) and the wheel radius ($0.5\ m$) is $Re = 5.19 \times 10^5$. The Froude number is $Fr = 0.05$.

The mesh is formed by 489313 tetrahedral elements and 109318 nodes. The time steps size used is 0.02 seconds.

In Figure 8 the evolution of the interface for different time steps is presented. For the first time step the whole domain is shown and for the remaining steps only the details at the wheel

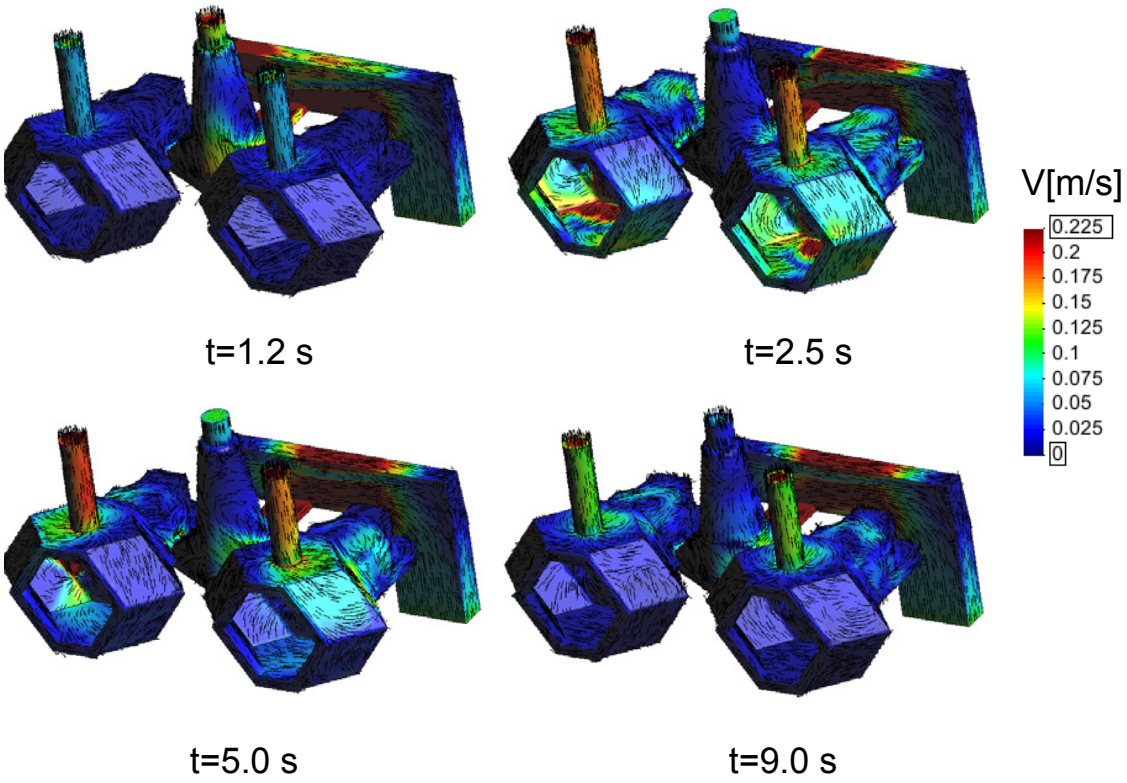

Figure 6: Velocity field at at $t = 1.2, 2.5, 5.0$ and 9.0 s using the fine mesh

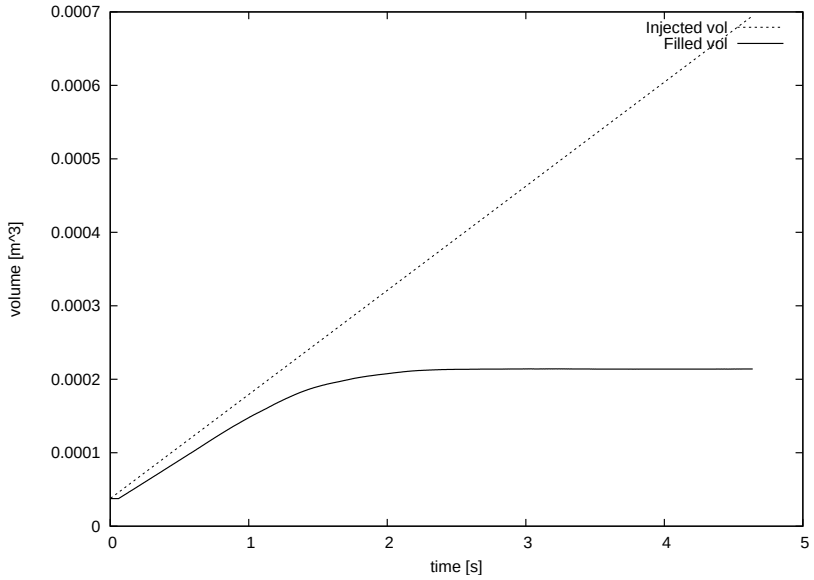

Figure 7: Filled volume vs. injected volume without using enrichment

are shown. It is interesting to see that at some points inside the spokes air is entrapped. This could lead to fabrication defects and should be avoided. At time step $t = 4.4\ s$ two air bubbles that are rising to escape as they reach the upper interface can be seen.

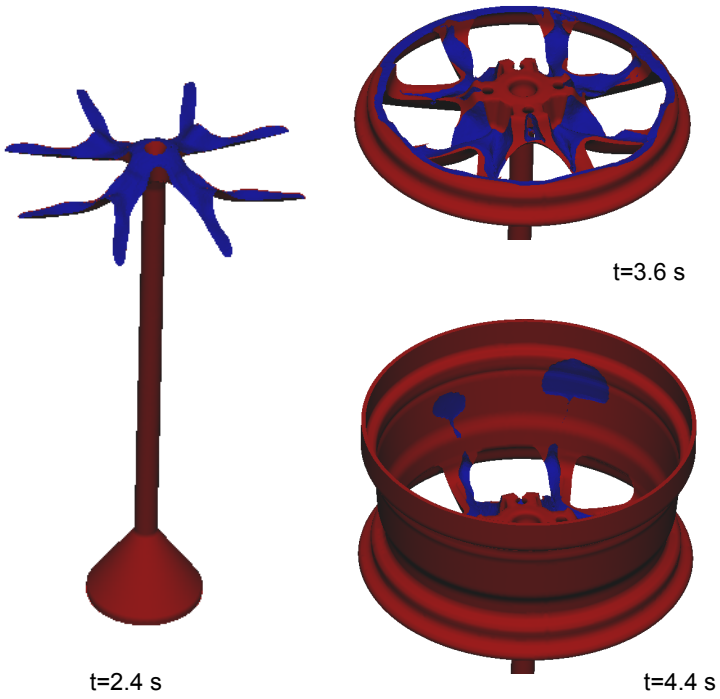

Figure 8: Interface evolution at t=2.4, 3.6 and $4.4\ s$

In Figure 9 the flow pattern for different time steps is presented. Since we are using an inlet pressure that varies linearly with time, while the interface is inside the filling tube the velocities remain quite constant. The increase in the inlet pressure is compensated mainly by an increase in the free surface height. Therefore the position of the free surface raises linearly with time and the velocity in the tube is approximately constant. As the flow enters the wheel and starts sliding down the wheel spokes the increase in the hydrostatic pressure stops but the inlet pressure continues growing linearly. Therefore the flow accelerates until the interface reaches the vertical walls. Finally, the flow rate stabilizes once again until the end of the simulation.

As in the previous example the simulation was also run without using the pressure enrichment. The results without pressure enrichment are much poorer than those obtained with pressure enrichment. In Figure 10 the evolution of the interface for the case without enrichment is shown. Up to $t = 2.4\ s$ the results are similar to those obtained with the enriched model. As the flow starts sliding down the wheel spokes the numerical mass loss becomes much more important than in the case with enrichment. For time $t = 3.6\ s$ the mass loss is easily noticeable and at $t = 4.4\ s$ it is very important.

5.3 Shovel

This example was presented to us as a really demanding case. It is the shovel for a power shovel. The filling process takes approximately half a minute and the shovel is nearly one meter long. The inlet velocity we have used during the simulation is $0.5\ m/s$. The Reynolds number

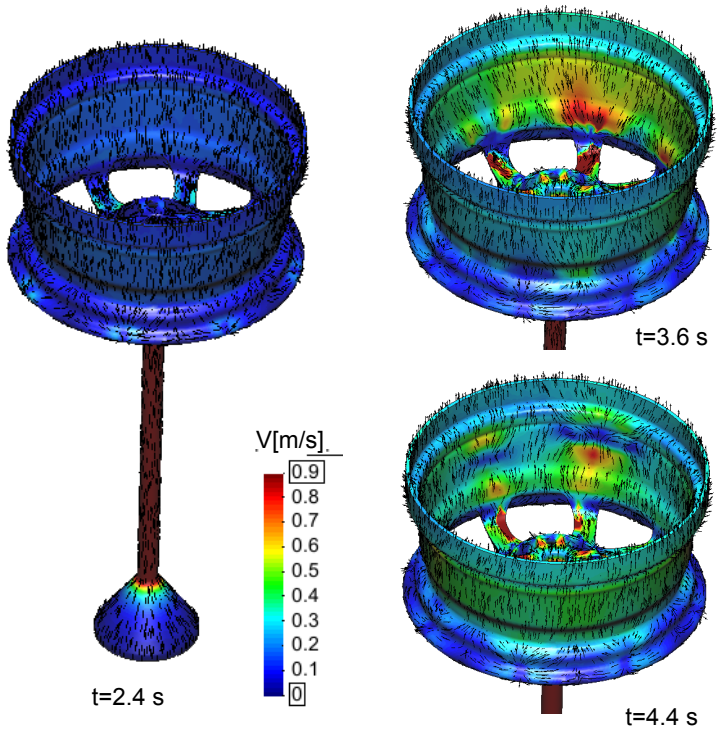

Figure 9: Velocity field at t=2.4, 3.6 and 4.4 s

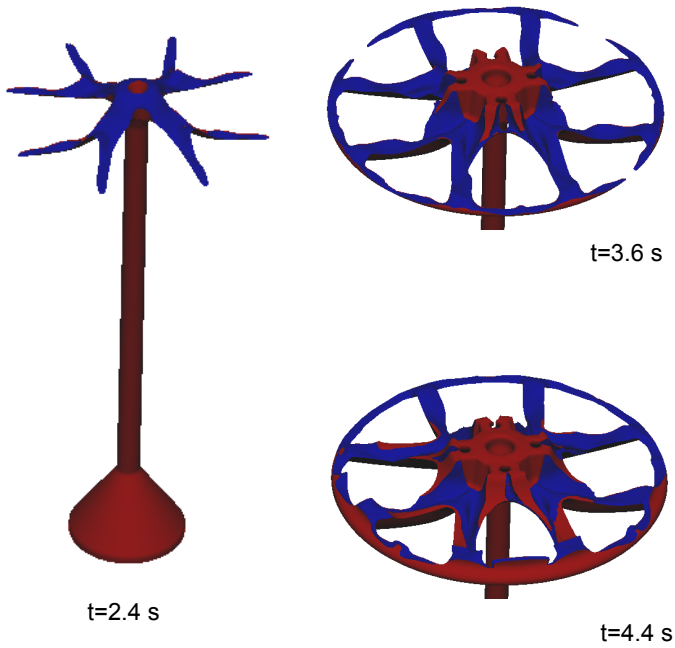

Figure 10: Interface evolution at t=2.4, 3.6 and 4.4 s for the case without pressure enrichment

based on the previous velocity and length is $Re = 4.44 \times 10^5$, and the Froude number is $Fr = 0.031$. As in the first example the material we have used is steel.

The mesh used for this example consists of 412848 tetrahedral elements and 87010 nodes. The time steps size is 0.02 seconds.

The filling channel used for this piece splits into two branches. One of the branches is closer to the inlet than the other one. As the interface reaches the first branch the molten metal starts flowing through it. Three seconds take place before the flow starts falling trough the second branch. Therefore the side of the shovel closer to the first branch is filled earlier than the part connected to the second branch. The position of the interface for selected time steps is shown in Figure 11.

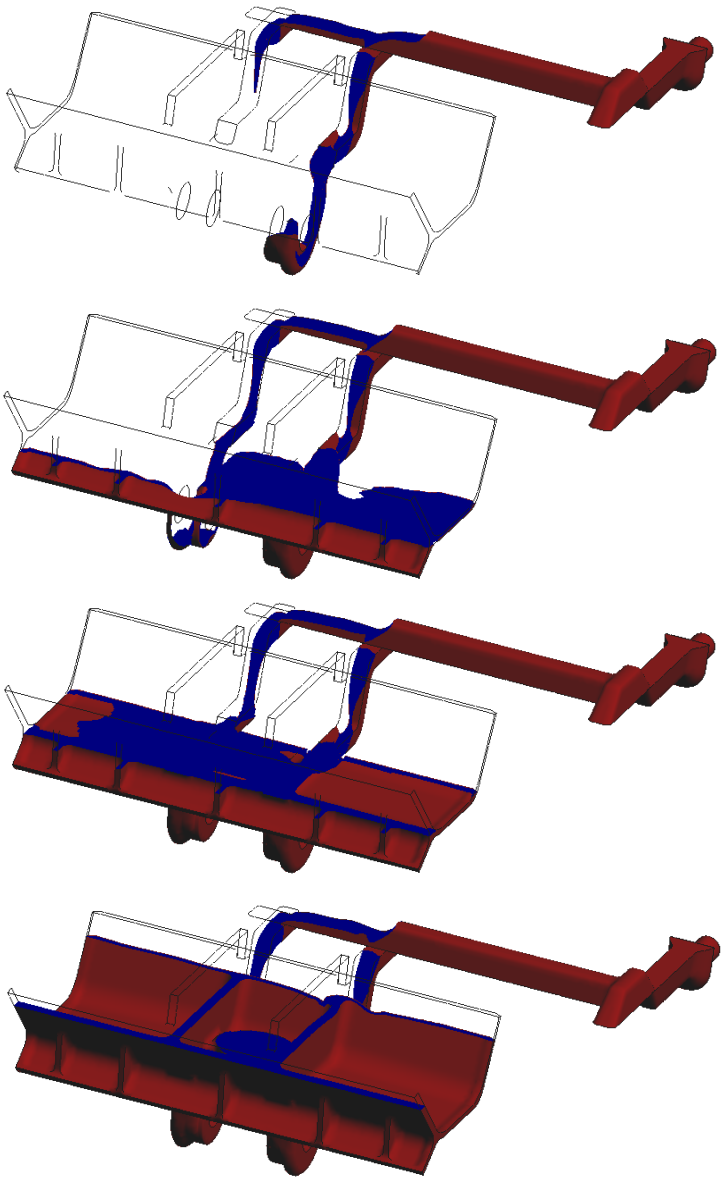

Figure 11: Interface position at t=3.3, 7.7 , 17.1 and 24.9 s

In Figure 12 we compare temporal evolution of the injected and filled volumes for the case

with pressure enrichment against the results obtained when no enrichment is used. As in the previous examples there is an important mass loss when no enrichment is used.

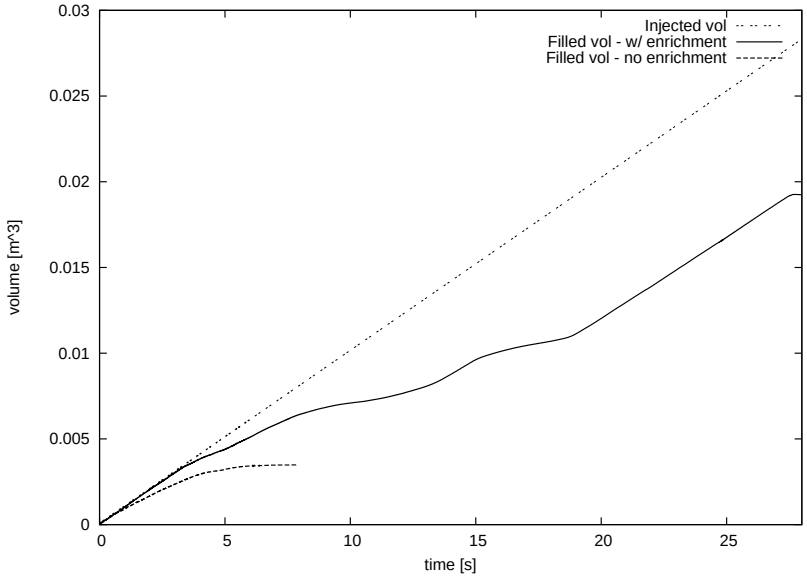

Figure 12: Filled volume vs. injected volume with and without pressure enrichment

6 CONCLUSIONS

In this paper we have presented a two phase flow model for mould filling problems. Is is specially suited for low Froude number flows. Such flows are quite common in foundry processes and three industrial examples have been presented. One of the key elements of the model is the enrichment of the pressure shape functions in elements cut by the front introduced in Coppola-Owen and Codina (2005). Here we have extended the use of the model to real world examples and shown that the pressure enrichment provides significant improvements for such flows.

The enrichment used is local to each element cut by the interface and can therefore be condensed prior to assembly, making the implementation quite simple on any finite element code. The computational overhead introduced by the enrichment and improved integration is small since it is only needed in cut elements.

REFERENCES

Brezzi F. and Fortin M. *Mixed and hybrid finite element methods*. Springer Verlag, 1991.

Chang Y., Hou T., Merriman B., and Osher S. A level set formulation of eulerian interface capturing methods. *Journal of Computational Physics*, 124:449–464, 1996.

Chessa J. and Belytschko T. A extended finite element method for two–phase fluids. *Journal of Applied Mechanics*, 70:10–17, 2003.

Codina R. A stabilized finite element method for generalized stationary incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 190:2681–2706, 2001.

Codina R., Schäfer U., and Oñate E. Mould filling simulation using finite elements. *International Journal of Numerical Methods for Heat & and Fluid Flow*, 4:291–310, 1994.

Coppola-Owen A. and Codina R. Improving Eulerian two-phase flow finite element approxi-

mation with discontinuous gradient pressure shape functions. *International Journal for Numerical Methods in Fluids*, 49:1278–1304, 2005.

Hirt C. and Nichols B. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39:201–225, 1981.

Launder B.E. and Spalding D.B. The numerical computation of turbulent flows. *Computer Methods in Applied Mechanics and Engineering*, 3:269–289, 1974.

Lewis R., Usmani A., and Cross J. Efficient mould filling simulation in metal castings by an explicit finite element method. *International Journal for Numerical Methods in Engineering*, 20:493–506, 1995.

Minev P., T.Chen, and K.Nandakumar. A finite element technique for multfluid incompressible flow using Eulerian grids. *Journal of Computational Physics*, 187:255–273, 2003.

Osher S. and Fedkiw R. Level set methods: and overview and some recent results. *Journal of Computational Physics*, 169:463–502, 2001.

Sussman M., Almgren A., Colella J., Howell L., and Welcome M. An adaptive level set approach for incompressible two phase flows. *Journal of Computational Physics*, 148:81–124, 1999.

Thompson E. Use of the pseudo-concentration to follow creeping viscous during transient analysis. *International Journal for Numerical Methods in Engineering*, 6:749–761, 1986.